AMENDMENTS TO CLAIMS

1. (Original) A method for muting a portion of an encoded bitstream of audio information comprising the steps of:

a) with respect to a current encoded audio frame of said encoded bitstream, computing a length of a dynamic template based on an error rate of said encoded bitstream, said dynamic template encompassing a plurality of previous encoded frames of said encoded bitstream;

b) summing errors of said plurality of previous encoded frames within said dynamic template to produce a first error sum;

c) determining if said first error sum exceeds a prescribed tolerance; and

d) adaptively merging muted error frames by muting said current encoded audio frame provided said first error sum value exceeds said prescribed tolerance whether or not said current encoded audio frame has an error.

2. (Original) A method as described in Claim 1 wherein said step a) comprises the steps of:

a1) with respect to said current encoded audio frame, summing errors of a plurality of previous encoded frames encompassed by a fixed-length template to produce a second error sum; and

a2) using said second error sum as an index to a look-up table to compute said length of said dynamic template.

3. (Original) A method as described in Claim 2 wherein said step a1) and said step b) are performed using an error array which contains a respective bit for each encoded audio frame indicating whether or not an error resides within its associated encoded audio frame.

4. (Original) A method as described in Claim 2 wherein said plurality of previous encoded frames encompassed by said fixed-length template are measured from and include said current encoded audio frame and wherein said plurality of previous encoded frames encompassed by said dynamic template are measured from and include said current encoded audio frame.

5. (Original) A method as described in Claim 4 wherein said fixed-length template is 24 audio frames in length and said tolerance is 1.

6. (Original) A method as described in Claim 2 wherein steps a2), b), c) and d) are bypassed if said second error sum is zero.

7. (Original) A method as described in Claim 2 wherein said encoded bitstream of audio information is substantially compliant with the AC3 digital audio standard.

8. (Currently Amended) A method for muting a portion of an encoded bitstream of audio information comprising the steps of:

a) detecting if a current encoded audio frame of said encoded bitstream contains an error; and

b) provided an error is detected, repeating a previous decoded audio frame in lieu of said current encoded audio frame, said ~~step b)~~ repeating comprising the steps of:

b1) obtaining decoded data of said previous audio frame;

b2) generating a repeated audio frame by replicating said decoded data of said previous audio frame for use in lieu of said current encoded audio frame;

b3) modifying said repeated audio frame by adding delay information of a last block of said previous audio frame with pulse code modulated (PCM) data of a first block of said repeated audio frame to generate new decoded data for said first block of said repeated audio frame; and

b4) sending said repeated audio frame to an audio output buffer for playout.

9. (Original) A method as described in Claim 8 wherein said step b1) obtains said decoded data from said audio output buffer.

10. (Original) A method as described in Claim 9 wherein said step b3) comprises the steps of:

shuffling and weighting said delay information to generate shuffled and weighted delay information;

weighting said PCM data to generate weighted PCM data;

adding said shuffled and weighted delay information with said weighted PCM data to generate said new decoded data for said first block of said repeated audio frame.

11. (Currently Amended) A method as described in Claim 9 further comprising the steps of:

c) provided an error is detected in a next encoded audio frame immediately following said current encoded audio frame, repeating said current encoded audio frame in lieu of said next encoded audio frame; said step c) comprising the steps of:

c1) obtaining decoded data of said current encoded audio frame;

c2) generating a second repeated audio frame by replicating said decoded data of said current encoded audio frame for use in lieu of said next encoded audio frame;

c3) modifying said second repeated audio frame by adding delay information of a last block of said current encoded audio frame with pulse code modulated (PCM) data of a first block of said second

repeated audio frame to generate new decoded data for said first block of said second repeated audio frame; and

[[b4)]] c4) sending said second repeated audio frame to an audio output buffer for playout.

12. (Original) A method as described in Claim 8 further comprising the steps of:

c) provided an error rate of said encoded bitstream is high, performing mute merging, said step c) comprising the steps of:

c1) with respect to said current encoded audio frame of said encoded bitstream, computing a length of a dynamic template based on an error rate of said encoded bitstream, said dynamic template encompassing a plurality of previous encoded frames of said encoded bitstream;

c2) summing errors of said plurality of previous encoded frames within said dynamic template to produce a first error sum;

c3) determining if said first error sum exceeds a prescribed tolerance; and

c4) adaptively merging muted error frames by muting said current encoded audio frame provided said first error sum value exceeds said prescribed tolerance whether or not said current encoded audio frame has an error.

13. (Original) A method as described in Claim 12 wherein said step c1) comprises the steps of:

with respect to said current encoded audio frame, summing errors of a plurality of previous encoded frames encompassed by a fixed-length template to produce a second error sum; and

using said second error sum as an index to a look-up table to compute said length of said dynamic template.

14. (Original) A method as described in Claim 8 wherein said encoded bitstream of audio information is substantially compliant with the AC3 digital audio standard.

15. (Withdrawn) In a digital decoder unit, a method for reducing audio frame over-run comprising the steps of:

a) receiving an audio encoded bitstream, decoding said encoded bitstream and storing decoded audio frames into an audio output buffer;

b) responsive to an audio mute signal, causing an audio bitstream of said decoder unit to zero, said step b) causing entries in said audio output buffer to zero starting from an entry position pointed to by a write pointer associated with said audio output buffer; and

c) directly zeroing a plurality of entries of said audio output buffer in response to said audio mute signal, said plurality of entries being a few entries away from a read pointer of said audio output buffer, said read pointer following

said write pointer and wherein as a result of step b) and step c), only a predetermined number of audio output frames are guaranteed to be played after said audio mute signal is detected.

16. (Withdrawn) A method as described in Claim 15 further comprising the step of d) freezing a current video frame on a display screen in response to said audio mute signal being detected and wherein said audio mute signal is generated as a result of a channel change command.

17. (Withdrawn) A method as described in Claim 15 wherein said read pointer and said write pointer are separated by a predetermined number of entries and wherein said audio output buffer is a circular buffer.

18. (Withdrawn) A method as described in Claim 17 wherein said predetermined number of entries is four and wherein said predetermined number of audio output frames is two.

19. (Withdrawn) A method as described in Claim 15 wherein said encoded bitstream is substantially compliant with the AC3 digital audio standard.

20. (Withdrawn) A method as described in Claim 15 wherein step c) is performed using a windowing function.